Amendment to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1-2 (canceled)

Claim 3 (currently amended): ~~The method of claim 2, further comprising~~ A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

identifying a second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set;

adding the new window to a map of the graphical user interface of the application;

analyzing the map to determine if the new window is already present in the map; and

adding a shortcut to the map if the new window is already present in the map, wherein the shortcut references the new window that is already present in the map.

Claim 4 (previously presented): A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

identifying a second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set;

analyzing the map to determine if the new window is already present in the map, wherein the new window is determined to already be present in the map if similarities between the new window and the window in the map are above a similarity threshold; and

adding the new window to a map of the graphical user interface of the application.

Claim 5 (original): The method of claim 4, wherein the similarity threshold is a percentage of graphical user interface objects that the new window and the window in the map have in common.

Claim 6 (currently amended): ~~The method of claim 2~~ A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

identifying a second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set;

adding the new window to a map of the graphical user interface of the application; and

analyzing the map to determine if the new window is already present in the map, wherein the new window is determined to already be present in the map if the new window and the window in the map have the same name.

Claim 7 (previously presented): A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

identifying a second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set;

analyzing the map to determine if the new window is already present in the map;

adding the new window to a map of the graphical user interface of the application; and

receiving input from a user that two or more windows of the map that have been determined as different should be considered the same window.

Claim 8 (previously presented): A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

identifying a second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set;

analyzing the map to determine if the new window is already present in the map;

adding the new window to a map of the graphical user interface of the application; and

receiving input from a user that two or more windows of the map that have been determined as the same should be considered different windows.

Claim 9 (currently amended): ~~The method of claim 1,~~ A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

identifying a second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set;

adding the new window to a map of the graphical user interface of the application; and

~~further comprising~~ receiving input from a user that one or more graphical user interface objects should be ignored when generating the map.

Claim 10 (previously presented): A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

receiving input from a user specifying an amount of time to wait after performing the action before identifying a second set of windows;

identifying the second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set; and

adding the new window to a map of the graphical user interface of the application.

Claim 11 (currently amended): ~~The method of claim 1,~~ A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

identifying a second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set;

adding the new window to a map of the graphical user interface of the application; and

~~further comprising~~ displaying the map on the computer.

Claim 12 (currently amended): ~~The method of claim 1,~~ A computer implemented method of mapping a graphical user interface of an application, comprising:

identifying a first set of windows that are active on the desktop of the computer;

performing an action on a graphical user interface object in a window of the application;

identifying a second set of windows that are active on the desktop of the computer;

comparing the first set of windows to the second set of windows to identify a new window in the second set; and

adding the new window to a map of the graphical user interface of the application, wherein the map is hierarchical and includes windows, graphical user interface objects and actions.

Claims 13-16 (canceled)

Claim 17 (previously presented): A system for testing applications, comprising:

an application mapper that programmatically executes an application to generate a map of the graphical user interface of the application, the application mapper adding a new window to the map by performing an action in the graphical user interface and identifying the new window by comparing windows in the graphical user interface before and after the action;

a script generator that utilizes the map to generate scripts that include instructions to test the application; and

an application tester that executes the scripts to test the application.

Claim 18 (original): The system of claim 17, wherein the application mapper generates the map by recursively performing actions on the graphical user interface of the application to identify new windows and adding the new windows to the map.

Claim 19 (original): The system of claim 17, wherein the map is hierarchical and includes windows, graphical user interface objects and actions.

Claim 20 (original): The system of claim 19, wherein the graphical user interface objects are buttons, sliders, check boxes, or tab controls.

Claim 21 (original): The system of claim 19, wherein the actions are left mouse click, right mouse click, left mouse double click, or keystrokes.

Claim 22 (cancel)

Claim 23 (currently amended): ~~The computer program product of claim 22,~~ A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that identifies a second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that adds the new window to a map of the graphical user interface of the application;

computer code that analyzes the map to determine if the new window is already present in the map;

~~further comprising~~ computer code that adds a shortcut to the map if the new window is already present in the map, wherein the shortcut references the new window that is already present in the map; and

a computer readable medium that stores the computer codes.

Claim 24 (previously presented): A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that identifies a second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that analyzes the map to determine if the new window is already present in the map, wherein the new window is determined to already be present in the map if similarities between the new window and the window in the map are above a similarity threshold;

computer code that adds the new window to a map of the graphical user interface of the application; and

a computer readable medium that stores the computer codes.

Claim 25 (previously presented): The computer program product of claim 24, wherein the similarity threshold is a percentage of graphical user interface objects that the new window and the window in the map have in common.

Claim 26 (currently amended): ~~The computer program product of claim 22,~~ A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that identifies a second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that adds the new window to a map of the graphical user interface of the application;

computer code that analyzes the map to determine if the new window is already present in the map, wherein the new window is determined to already be present in the map if the new window and the window in the map have the same name; and

a computer readable medium that stores the computer codes.

Claim 27 (previously presented): A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that identifies a second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that analyzes the map to determine if the new window is already present in the map;

computer code that adds the new window to a map of the graphical user interface of the application;

computer code that receives input from a user that two or more windows of the map that have been determined as different should be considered the same window; and

a computer readable medium that stores the computer codes.

Claim 28 (previously presented): A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that identifies a second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that analyzes the map to determine if the new window is already present in the map;

computer code that adds the new window to a map of the graphical user interface of the application;

computer code that receives input from a user that two or more windows of the map that have been determined as the same should be considered different windows; and

a computer readable medium that stores the computer codes.

Claim 29 (currently amended): ~~The computer program product of claim 15,~~ A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that identifies a second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that adds the new window to a map of the graphical user interface of the application;

~~further comprising~~ computer code that receives input from a user that one or more graphical user interface objects should be ignored when generating the map; and

a computer readable medium that stores the computer codes.

Claim 30 (previously presented): A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that receives input from a user specifying an amount of time to wait after performing the action before identifies a second set of windows;

computer code that identifies the second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that adds the new window to a map of the graphical user interface of the application; and

a computer readable medium that stores the computer codes.

Claim 31 (currently amended): ~~The computer program product of claim 15,~~ A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that identifies a second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that adds the new window to a map of the graphical user interface of the application;

~~further comprising~~ computer code that displays the map on the computer; and

a computer readable medium that stores the computer codes.

Claim 32 (currently amended): ~~The computer program product of claim 15,~~ A computer program product for mapping a graphical user interface of an application, comprising:

computer code that identifies a first set of windows that are active on the desktop of the computer;

computer code that performs an action on a graphical user interface object in a window of the application;

computer code that identifies a second set of windows that are active on the desktop of the computer;

computer code that compares the first set of windows to the second set of windows to identify a new window in the second set;

computer code that adds the new window to a map of the graphical user interface of the application, wherein the map is hierarchical and includes windows, graphical user interface objects and actions; and

a computer readable medium that stores the computer codes.

Claims 33-34 (canceled)